Journal of Biomedical Informatics 43 (2010) 694–700

Contents lists available at ScienceDirect

Journal of Biomedical Informatics

journal homepage: www.elsevier.com/locate/yjbin

Reflective random indexing for semi-automatic indexing of the biomedical literature

Vidya Vasuki [a,*], Trevor Cohen [b]

[a] Center for Decision Making and Cognition, Department of Biomedical Informatics, Arizona State University, Arizona, USA
[b] Center for Cognitive Informatics and Decision Making, School of Health Information Sciences, University of Texas, Houston, USA

ARTICLE INFO

Article history:
Received 11 August 2009
Available online 9 April 2010

Keywords:
Indexing
MeSH terms
MEDLINE indexing
Reflective Random Indexing
Random Indexing
Information retrieval
Distributional semantics
Text categorization

ABSTRACT

The rapid growth of biomedical literature is evident in the increasing size of the MEDLINE research database. Medical Subject Headings (MeSH), a controlled set of keywords, are used to index all the citations contained in the database to facilitate search and retrieval. This volume of citations calls for efficient tools to assist indexers at the US National Library of Medicine (NLM). Currently, the Medical Text Indexer (MTI) system provides assistance by recommending MeSH terms based on the title and abstract of an article using a combination of distributional and vocabulary-based methods. In this paper, we evaluate a novel approach toward indexer assistance by using nearest neighbor classification in combination with Reflective Random Indexing (RRI), a scalable alternative to the established methods of distributional semantics. On a test set provided by the NLM, our approach significantly outperforms the MTI system, suggesting that the RRI approach would make a useful addition to the current methodologies.

1. Introduction

The MEDLINE database, maintained by the US National Library of Medicine (NLM) is the most comprehensive source of biomedical bibliographic information in existence. As of March 2009, MEDLINE contains over 17 million citations and the number continues to grow steeply, with over 450,000 citations added in 2008 alone [1]. In order to index, search and catalog these citations, the NLM employs a controlled terminology, the Medical Subject Headings (MeSH). The 2009 edition of the MeSH contains 25,186 main headings and 83 sub-headings [2] and this vocabulary grows with the introduction of new concepts. The task of assigning MeSH terms to new citations is labor intensive, and the development and evaluation of automated approaches to assist with this task have been the subjects of considerable research. Much of this research [3–5] has been conducted under the auspices of the NLM's Indexing Initiative, which has produced the Medical Text Indexer (MTI) system [6]. The MTI automatically generates ordered lists of MeSH suggestions and is currently used by human curators at the NLM as an assistive tool.

In this paper, we evaluate the use of Reflective Random Indexing (RRI) [7], an iterative variant of the Random Indexing (RI) method [8] in combination with nearest neighbor search, a method that has been well-researched and successfully employed for classification purposes [9], as a basis for automatic assignment of MeSH terms to MEDLINE citations. The primary motivation of this paper is to determine the extent to which these emerging and scalable methods of distributional semantics are applicable to the problem of semi-automated indexing. However, the paper also has broader implications for information retrieval as it provides further support for the utility of RRI as a scalable solution to the problem of information retrieval on the basis of conceptual content rather than specific word choice.

2. Background

2.1. Approaches to semi-automated indexing

Semi-automated indexing approaches seek to ease the task of the human indexer by providing appropriate MeSH term suggestions. Considerable research attention has been devoted to this issue. While the research focus has recently shifted to MeSH main heading/subheading pair suggestions for finer grained indexing, attaching sub-headings still depends on the stand-alone main heading suggestions, which are the focus of this paper. Approaches to this problem have, in general, been either distributional or vocabulary-based in nature.

Distributional methods are based on the statistical distribution of terms and previously assigned categories in a labeled training set. These include (i) The Expert Network approach [10], in which previously indexed documents are represented as points in a high-dimensional document-by-term space and MeSH term assignment

* Corresponding author. Address: Center for Decision Making and Cognition, Department of Biomedical Informatics, Arizona State University, 15827 NE Leary Way Apt B107, Redmond, WA 98052, USA.
E-mail address: VidyaVasuki@gmail.com (V. Vasuki).

1532-0464/$ - see front matter © 2010 Elsevier Inc. All rights reserved.
doi:10.1016/j.jbi.2010.04.001

is based on the labels assigned to the nearest neighbors of an unseen query document; (ii) Linear Least Squared Fit (LLSF) [11], a regression method which learns term to category associations from a set of labeled training data; (iii) the Pindex program [12] which uses the conditional probability of a particular phrase occurring given a particular MeSH term to map text to categories; and (iv) Naïve Bayesian classification [3,5], a statistical machine learning method. Vocabulary-based methods draw on domain-specific knowledge resources such as the MeSH [2] and the UMLS Metathesaurus [15], a vocabulary database that contains biomedical and health-related concepts, synonyms and relationships between them. These methods, which include MetaMap indexing [13], Trigram indexing [14] and (iv) String Matching (STR) [10] assign categories based on surface similarities between the citation text and descriptors or concepts in the vocabulary resource. The methods used to estimate the extent of this similarity vary, and may be statistically motivated as in the case of the Trigram method.

Since all the methods discussed above have not been tested on a common test set and do not use the same evaluation metrics, it is difficult to draw a comparison between them based on the published literature. To address this issue, the NLM have provided a test set of 200 abstracts [16] to encourage research in the area and enable meaningful comparison between methods moving forward.

2.2. The Medical Text Indexer (MTI)

Since 2002, the NLM indexers have been using the MeSH main heading recommendations generated by MTI. The MTI system [17], developed by the NLM, uses a combination of distributional and vocabulary-based approaches to recommend indexing terms (MeSH descriptors) for a given citation title and abstract. This section outlines the process flow of the system.

There are two parallel paths in the flow, each of which starts with the text from the title and abstract of the citation that needs to be indexed as input, and generate a ranked list of MeSH descriptors.

1. MetaMap Indexing – The PhraseX program extracts noun phrases from the citation text and MetaMap maps these phrases to candidate medical concepts using the UMLS Metathesaurus. This is done to ensure that the concepts that are referenced in the text are unique. The candidate UMLS concepts are then ranked and restricted to MeSH descriptors using synonyms, associated expressions and inter concept relationships.
2. PubMed Related Citations – This is a statistical approach where citations are represented using the bag-of-words model. Local and global schemes are used to weight the words. Related citations are obtained by computing the similarity between citations. Similarity is measured by sum of weights (local1 $\times$ local2 $\times$ global) of the words common between the two citations. Finally, the MeSH descriptors used to index these related citations are ranked.

The last step in the MTI indexing flow combines and ranks all the MeSH headings generated as the output of each of these methods. This step also involves extensive post-processing embodying NLM indexing policy.

On the same test set containing 200 citations that is used for evaluating the performance accuracy of our system, the MTI produced 14–31 MeSH main heading recommendations per citation with an overall precision of 0.3352, an overall recall of 0.5593 and an overall F_1-measure of 0.4192.

2.3. Distinguishing features of our approach

Of these methods, both the Expert Network and the PubMed Related Citations approaches [10] are particularly pertinent to the method used in this paper. In these approaches, citations are represented as vectors with a dimension corresponding to each term in the text corpus according to Salton's classic vector space model for information retrieval [18]. MeSH recommendations are generated on the basis of the similarity between the vector representation of an unlabeled citation and those of the nearest-neighboring pre-indexed citations in the space. However, our approach differs from Expert Networks in several respects. Firstly, unlike the full document-term matrix used by this method, our approach reduces the dimensions of this space using RI. RI presents a scalable alternative to established distributional methods such as Latent Semantic Analysis (LSA) [19], to derive a condensed vector representation for each document. Unlike LSA and other methods that depend on computationally demanding Singular Value Decomposition (SVD), dimension reduction in this case is done on the fly, without constructing the full term-document matrix. This provides the advantage of efficiency in computation and storage. However, as the original implementation of RI is limited in its ability to derive meaningful connections between terms that do not co-occur directly in the same document [7], we use RRI, an iterative variant of the original model that has been customized for this purpose. This retains the scalability advantages of RI, but also allows for mapping between similar documents which do not share any common terms, in a manner similar to LSA. This ability to identify similarities between documents on the basis of conceptual content rather than specific terms further distinguishes our approach from Expert Networks and PubMed Related Citations, and is also a feature of the LSA approach to automated grading of content-based essays [20]. This approach, which involves the assignment of scores to ungraded essays based on the grades assigned to neighboring graded essays in a reduced-dimensional LSA space, was the immediate inspiration for our approach.

2.4. Random Indexing (RI)

2.4.1. Mathematical foundations

RI [8,21,22] is a stochastic technique that involves projecting term vectors into a low dimensional space (relative to the number of documents in the corpus). Underlying RI is the observation that although a d-dimensional space can have only d perpendicular axes, it can contain many more "nearly orthogonal" axes. In RI, these are constructed by distributing a small number of +1 and -1 values, called the seeds, across a k-dimensional random index vector, where $k < d$ [8]. This number of seeds is called the "seed length". The other values in this vector are zero, and k is usually on the order of 1000. Owing to the sparseness of the random index vectors, there is a high probability of these being close to orthogonal to one another – their relatedness as measured using the commonly employed cosine metric is likely to be close to zero. Consequently, the semantic relatedness measured in the reduced-dimensional matrix approximates that in the fully orthogonal term-document matrix, conserving storage space and allowing similarity computation at a fraction of the computational cost.

RI and related methods such as Random Projection [23] are supported by the Johnson–Lindenstrauss lemma, which gives the upper bound on the error introduced by projecting points in a vector space into a uniform, random lower-dimensional subspace [24]. It follows from the lemma that the distance between points will be approximately preserved with high probability if they are projected into a reduced-dimensional random subspace of sufficient dimensionality. This relationship between error and dimensionality as well as other applications of the RI method are discussed in detail in [23].

2.4.2. Random Indexing, as originally implemented

The original implementation of RI starts by assigning a sparse random index vector to every document in the corpus. The index vector for each document a given term occurs in is added to produce a semantic vector for each term [21]. The computational advantages of using this method of document representation are twofold. One, we do not need to represent the full term-document matrix. Two, dimension reduction is done on the fly by simple vector addition, rather than using computationally expensive methods like Singular Value Decomposition which is used in LSA. However, the drawback of this approach is that, documents that do not have any words in common but contain synonyms of the same concept are likely to be represented as nearly orthogonal vectors. This limitation of RI is discussed in detail and shown empirically in [7] and follows logically from the observation that synonyms seldom occur in the same context. As the vector representation for any given term is the linear sum of the near-orthogonal document vectors for the documents it occurs in, the vector representations for two terms that do not co-occur together in any document will be almost orthogonal to one another. With respect to information retrieval, this means that the model is limited in its ability to retrieve relevant documents that do not contain one of the cue terms in a query. With regard to semi-automated indexing, this limits the ability of the model to find meaningful similarities between related documents that do not contain any common terms.

2.4.3. Reflective Random Indexing (RRI)

In this research, we use RRI, an iterative variant of the RI approach shown to be better able to draw meaningful associations between terms that do not occur in the same document [7]. In RI, a document is represented as the weighted sum of vector representations for the words that it contains, and each term in the corpus is composed of the linear sum of the vectors for the documents that it appears in. It has been observed [7,25] that it is possible to train the term and document vectors cyclically using this process, as shown in Fig. 1. Such retraining has been found to improve the ability of RI to make "indirect inferences" [26], drawing meaningful associations between terms that do not co-occur directly in any document. For example, words such as "doctor" and "physician" which initially may have been represented by nearly orthogonal index vectors will be represented as two vectors close to one another as a result of the RRI process. This effect on individual terms will be extended to entire documents: the vector representations for citations containing the word "doctor" will be similar to those containing the word "physician", when document vectors are constructed from term vectors produced by the RRI process.

The source code of the Semantic Vectors Package [27], an open source implementation of RI using Apache Lucene [28], forms the basis for the computational implementation used in this work. Both the authors are active contributors to this package, which in its latest iteration includes an implementation of the variant of RRI employed in this research.

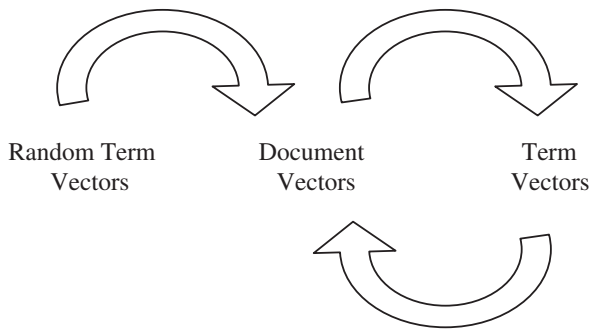

Fig. 1. Cyclical training using RRI.

3. Evaluation

In this section, we evaluate the utility of the RRI approach as a basis for semi-automated indexing. We use a nearest-neighbor approach similar in nature to that employed by the Expert Networks system [10] and PubMed Related Citations [6], as well as in automated grading of content-based essays using LSA [20]. The idea underlying these approaches is that within a vector space, the human-assigned labels on the nearest-neighboring documents to a cue document are likely to be applicable to this cue document also.

3.1. Methods

3.1.1. Training of the model

The system is trained using all the 9,003,611 citations that contain abstracts and have already been labeled with MeSH terms, available in the 2007 baseline release of MEDLINE minus the set of 200 MEDLINE citations provided by NLM [16], which are used as the test set. There is no overlap between the test and training sets. The number of words included in the model is reduced by using global term frequency thresholds. An empirical upper bound derived by inspecting the distribution of terms in the corpus is used to filter out words with high frequency that do not contribute towards the semantics of the document. These could include, for example, articles or prepositions. Infrequently occurring words below the lower threshold of 10, which include occasional spelling mistakes, tokenization errors and terms referring to concepts that are minimally represented in the database are not considered either. Of the remaining words, only those that contain at least one letter and not more than three non-alphabet characters are taken into account. This allowance for non-alphabet characters helps to retain valid medical terms like gene and enzyme names.

To generate the vector representation for documents and terms in the training set, random index vectors are assigned to each term under consideration. The dimension of the vectors varies across runs from 500 to 1500. Since sparse vectors are required to ensure near-orthogonality, a small seed length on the order of 20 has been used. This seed length has been used successfully in prior applications of RI, and appears adequate to preserve meaningful associations between terms in the reduced-dimensional space. Preliminary document vectors are obtained by computing the linear sum of the random term vectors for each term in a given document. As in the original implementation of RI, vector addition in RRI is frequency weighted: if a term occurs twice in a document, the random index vector for this term is added to the document vector twice, and so forth. At this stage, the vectors for two documents that cover similar content without containing any common terms should be close to orthogonal to one another. For each term, frequency-weighted vectors for the documents in which the term occurs are summed and normalized (i.e., their length is truncated to 1) to generate a set of semantic term vectors. Documents are then retrained using these semantic term vectors as the basis vectors, resulting in a set of semantic document vectors, which are then normalized and used as the training set. This process is shown in Fig. 2.

In certain runs, rather than using raw frequency weighting, terms are weighted using the log-entropy weighting scheme while computing the document vectors. This has the beneficial effects of reducing the effect of extreme differences in local frequency of terms by giving greater emphasis to terms that are likely to refer to specific concepts in the corpus. The formula used to compute the log-entropy weight of each term in the corpus is shown below.

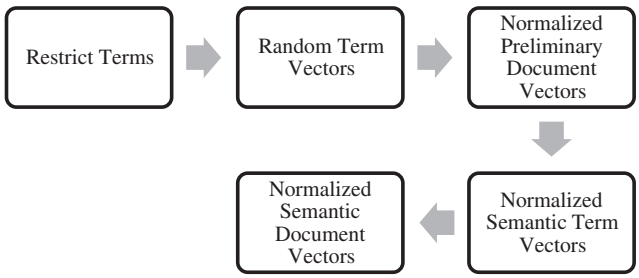

Fig. 2. Stages of vector construction in RRI.

This weighting scheme is employed in LSA and has been shown to be beneficial in a number of applications [29].

$$\text{Global weight } (i) = 1 + \sum_j \frac{p_{ij}\log_2(p_{ij})}{\log_2 n} \quad \text{where,}$$

$$p_{ij} = \frac{\text{Frequency of term } i \text{ in document } j}{\text{Global frequency of term } i}$$

$$\text{Local weight } (i,j) = \log(1 + \text{Frequency of term } i \text{ in document } j)$$

$$\text{Log entropy } (i,j) = \text{Global weight } (i) \times \text{Local weight } (i,j)$$

At the end of the RRI phase, documents and terms are represented in a k-dimensional vector space, where 'k' is the dimensionality of the pre-assigned random index vector. Each unseen abstract from the test set is also represented in this manner by computing the linear sum of the semantic term vectors for the words that it contains in this vector space, and normalizing this vector.

3.1.2. Ranking of MeSH candidates

This step of the indexing process resembles the Pubmed Related Citations algorithm employed by the MTI [6], and the Expert Networks approach [10]. To obtain the MeSH recommendations, the k-nearest-neighboring documents of the test document are retrieved from the training set. Cosine similarity (the scalar product of normalized document vectors) is used to measure the likeness between vectors. Human indexed MeSH terms that index these neighboring documents are candidates for recommendation. Each such candidate is assigned a fitness score which is the sum of the similarities between the documents it is associated with and the test document. This process is shown in Fig. 3: if a particular test document (Test doc) has two nearest neighbors that are pre-indexed with the MeSH term "Atrophy", the fitness score for Atrophy will be the sum of the cosine similarities of each of these two k-nearest neighbors to the cue document. Only the top 25 among the ranked list of candidates are chosen.

In order to ensure that a recommended MeSH term is reasonably frequent among the neighbors, only MeSH terms with a fitness score above 1 are considered. This cutoff mandates that every MeSH term that is recommended is used to index more than one neighboring abstract, which increases the overall precision, with little trade-off in recall.

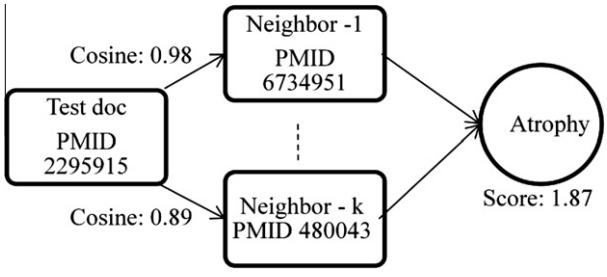

Fig. 3. Scoring MeSH candidates.

3.1.3. Evaluation metrics

Prior evaluations of MeSH term assignment quality have considered human judgments as a gold standard, either by comparing with human-assigned MeSH categories, or by testing the performance of a retrieval algorithm based on human classification of documents as relevant or irrelevant. We follow the prior method of evaluation in this study. Precision and recall values are computed based on human categorization for each of the test abstracts and are averaged over the set. These values are combined by computing the F_1-measure, the weighted harmonic mean of precision and recall:

$$F_\beta\text{-measure} = \frac{(1+\beta^2)\cdot \text{precision} \cdot \text{recall}}{\beta^2 \cdot \text{precision} + \text{recall}}$$

where, $\propto$ and β are the weights that denote the importance assigned to precision and recall, respectively. Although the choice of these values is debatable for this application, we have chosen the F_1-measure (F-measure) that gives equal importance to both for easy comparison with previous results. This is calculated using the above formula with $\beta = 1$. In addition, we include the F_2-measure, which is calculated using the above formula with $\beta = 2$, as it has been suggested that professional indexers may value recall over precision, and the F_2-measure weights these metrics accordingly. Since the number of MeSH categories that will be associated with an abstract is not known, the top 25 in the ranked list of candidate terms are suggested to the indexer. In addition to this, the use of a minimum threshold of 1 for MeSH score is also evaluated. This is to ensure that weakly associated candidates do not reduce the precision.

3.2. Results and discussion

3.2.1. Results

Using a 500 dimensional space and 10 nearest-neighboring abstracts, we generated overall performance statistics (Table 1) in a manner consistent with that used to generate the published MTI results. The overall precision and recall values are calculated from the total number of correct, incorrect and missed recommendations generated by the system across all the 200 test documents. The overall F_1- and F_2-measures are computed from these cumulative values.

The best of these results exceeds the MTI results published by NLM [16]. With a MeSH cutoff of 1, the number of suggestions made for same test documents is kept below 25 resulting in an increase in precision with a small compromise in the recall. It is also observed that the use of log-entropy weighting results in better suggestions. Also, using just the MeSH cutoff without log-entropy weighting for terms also shows results better than those produced by MTI. The ability to achieve this performance without using log-entropy weighting is significant because it preserves the desirable property of incremental updates to the model.

In addition, we analyzed the average precision, recall and F-measures across documents, and observed similar improvements

Table 1
Results with 500 Dimensions and 10 Neighbors, with MTI results provided as a point of comparison. * is used to indicate the best run. (LE – log-entropy).

MeSH cutoff	Weight	Precision	Recall	F_1-measure	F_2-measure
0	LE	0.3271	0.598	0.4229	0.5131
1	LE	0.3723*	0.575*	0.4519*	0.5185
0	None	0.3222	0.5893	0.4166	0.5055
1	None	0.3741	0.5633	0.4496	0.5115
MTI		0.3352	0.5593	0.4192	0.4933

over MTI with all metrics. Improvements in precision and both F-measures were statistically significant (pairwise t-test, $p < 0.05$).

3.2.2. Scalability

Training the system using RRI involves building vector representations for documents and terms in the corpus. This process (essentially vector addition) scales at a rate that is linear to the size of the corpus. During the training phase, the document vectors are read from and written to the disk in a sequential manner, while holding only the term vectors in RAM. Consequently, the space requirement for training the system using RRI is on the order of Tk, where T is the number of terms in the corpus and k is the number of dimensions in the random subspace. More details on the underlying algorithms can be found in [7]. Calculating document vectors using log-entropy weighting for the set of 9,003,411 abstracts used in the training took around 3.5 h on a 64-bit server with 16 Gb of RAM. With the current implementation, performing a nearest neighbor search is the most time consuming aspect of generating MeSH recommendations for a new document. The search implementation used in this research walks through the full set of document vectors on disk without optimizing this step, which may be necessary in the production environment.

3.2.3. Procedure for incremental updates

In order to achieve incremental updates, the semantic term and document vectors are not normalized prior to storage, and the random index vectors used to build the training set are retained to facilitate the update process.

By not weighting the terms using log-entropy weighting, it is possible to immediately make use of every abstract that is added to the database for training. The procedure to update the system on addition of new labeled abstracts is shown in Fig. 4. The update process consists of three steps. (i) Whenever a new abstract is indexed with MeSH terms, all the random term vectors for the words contained in it are retrieved from the training set repository. Any terms in the document that are not yet represented by random index vectors are assigned sparse random vectors, which are stored. The normalized sum of these vectors gives the preliminary vector for the new document. (ii) This new vector is added to each of the semantic vectors in the repository for the words contained in the new document. (iii) The semantic vectors for the documents affected by the change in the term vectors are recomputed and updated. This is accomplished by adding the new preliminary document vector multiplied by the number of words common between the document being updated and the newly added document. (iv) Since the semantic term vectors are not stored as unit vectors (vectors of length one), they are normalized before computing their linear sum, which gives the final semantic vector representation of the new document. This new document vector is then added to the semantic document vector repository, and will consequently be considered for classifying unseen abstracts that follow. Incremental update is a unique advantage of the RI approach, and is not possible with methods such as LSA or LLSF.

3.2.4. Size of the training set

The size of the training set plays a key role in the success of this approach by improving the quality of MeSH recommendations. A large training set translates to more pre-indexed examples for each MeSH category available to learn the association from. Also, more samples to draw term-document correlation from results in better vector representation for documents and terms. The advantage obtained from the capability of the method to harness all of the available indexed data is sustained by incrementally adding new documents to the training store. The positive effect of the size of the training set on the quality of the recommendations is illustrated in Fig. 5.

3.2.5. On latent semantics

The primary advantage of the reflective variant of RI used in this paper is its ability to provide a scalable means of measuring the similarities in meaning between passages of text regardless of the specific words used to express this meaning. This ability to represent the meaning underlying the choice of words used to express it, the "latent" semantics of a passage, was the primary motivation for the use of LSA in information retrieval [19], as it supports the retrieval of relevant documents that do not contain the specific terms used in a query.

In order to investigate the effect of reflection (and therefore latent semantics) on the performance of our system, we perform the nearest neighbor evaluation using the preliminary document vectors. A preliminary document vector for a given document is constructed as the linear sum of the near-orthogonal random index vector for each term occurring in this document. Consequently, the vector space constructed as a result of this process is a reduced-dimensional approximation of the original Salton vector space model. To the extent that the distance between points is accurately preserved through the dimension reduction process, the vector representations for documents that do not share any common terms will be almost orthogonal and the similarity between them as measured with the commonly used cosine metric will be close to zero.

As shown in Table 2, the precision, recall and F_1-measure are significantly improved when the RRI approach is used. This finding is consistent with our previous research [7], which shows that RRI is better able than RI to predict future co-occurrence of terms in

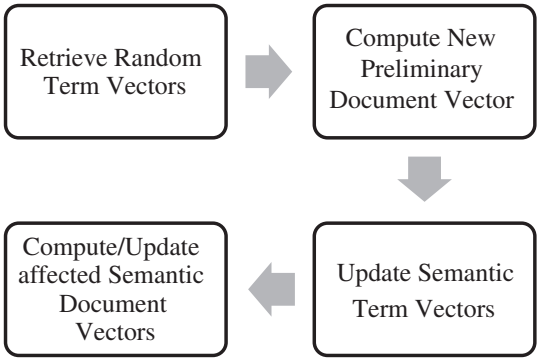

Fig. 4. Incremental update process.

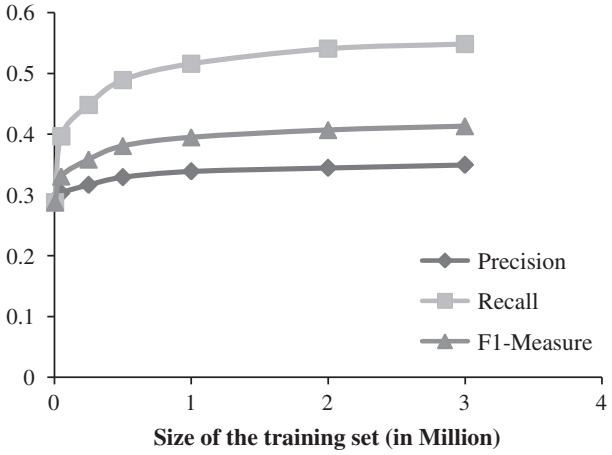

Fig. 5. Effect of the size of training data on the quality of MeSH terms.

Table 2
Difference in performance with and without reflection on the NLM test set. These results use 500 dimensional vectors, MeSH cutoff = 1 and log-entropy term weighting. All of the differences are statistically significant (pairwise t-test, $p < 0.05$).

N = 200 citations	Without reflection	With reflection
Mean precision	0.2909	0.3684
Mean recall	0.5479	0.5894
Mean F_1-measure	0.3691	0.4419

MEDLINE from a time-delimited training set. As terms that co-occur directly in the future (but not in the training set) are likely to be meaningfully related, these studies provide empirical support for the ability of RRI to represent latent semantics. The improvement in performance in the semi-automatic indexing task suggests that the latent semantics provided by RRI are important for this application. Interestingly, this improvement in performance with reflection was not evident in an evaluation using the Test of English as a Foreign Language (TOEFL) synonym test [30], which evaluates the ability of a model to correctly select a synonymous term to a cue term from a selection of four possibilities. As examination of the TOEFL test set shows the correct answer co-occurs in the same document with the cue term in the majority of cases, we have some reservations its sensitivity as a measure of the ability to find meaningful connections between terms that do not co-occur.

3.2.6. Parameter selection

In order to determine the optimal dimension for vector representation, the influence of dimensionality for this task is evaluated. The evaluation of the accuracy of recommendations generated by 100, 500, 1000 and 1500 dimensional vector space (Fig. 6) suggests an optimal dimensionality of 500. Lower dimensionality reduces the computational and space requirements of the process.

A simple experiment to observe the effect of the number of neighboring documents selected on the quality of MeSH suggestions shows the best choice to be 10 neighbors. The precision, recall and F_1-measure trends obtained are shown in Fig. 7.

3.2.7. Error analysis

Analysis of the errors during system performance shows that false positive errors involve more frequently occurring MeSH headings than false negative errors. Also, the averages of the term frequencies of each of these sets are higher than the average across all MeSH terms. This suggests that the distributional statistics of MeSH terms themselves may be of use in improving system performance. We note, however, that evidence exists that considerable inconsistency exists within the human ratings that are included in our training set [31], which ultimately limits the accuracy that can be obtained when these ratings are used as a gold standard.

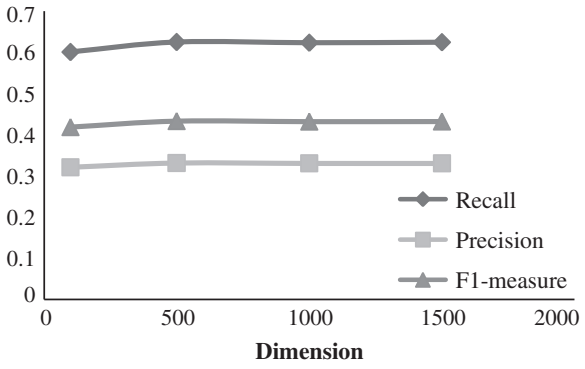

Fig. 6. Accuracy of MeSH suggestions for 10 neighbors and no MeSH cutoff.

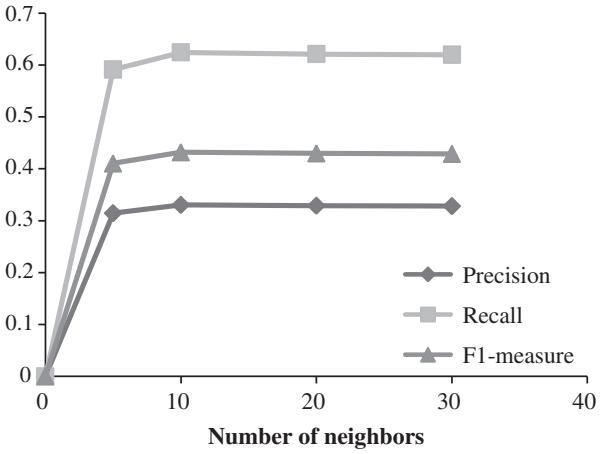

Fig. 7. Accuracy of MeSH suggestions for 1500 dimensions and no MeSH cutoff.

4. Implications

The immediate implication of this work is the addition of another promising method to the set of tools available to assist indexers with their task. The performance of this method relates to the volume of data it is able to learn from. With a scalable implementation, this learning process could continue, as new indexed abstracts are added to the system. The idea of determining human-assigned categories for a new item in a set from the nearest-neighboring items with pre-assigned categories is not in itself new. However, the implementation presented here differs from prior approaches in its ability to derive meaningful connections between documents with no shared terms in a scalable manner. This scalable approach to latent semantics has broad implications, which have now been demonstrated in the context of semi-automated indexing and literature-based knowledge discovery. The RRI method, as implemented in this research, can be applied to almost any single class or multi-class text categorization problem that provides a set of pre-classified samples for training purposes. The performance result of this system on the MEDLINE indexing dataset stands as one example of such an application. However, we anticipate further application of these, and related methods to the ever-expanding data sets that characterize the information era.

5. Limitations and future work

The evaluation performed in this study uses a training set far larger than the test set. While this is certainly not the norm in the machine learning literature, it does prove the scalability of the method and is also arguably an ecologically valid evaluation as it is consistent with the amount of training data available in the actual indexing environment, more so with the use of incremental system updates. In addition, as the RRI algorithm begins with random initiation of document vectors, performance would be expected to vary slightly across runs. Although we have not observed any statistically significant changes in precision, recall or F-measure when retraining with a different set of random index vectors, more research is needed to determine to what extent performance fluctuates across repeated runs. The use of distributional weighting of MeSH recommendations may be explored as an avenue to further improve performance.

6. Conclusion

This research explores the application of a novel approach based on RRI to predict MeSH terms for indexing MEDLINE citations. The best results obtained by this approach outperform the results generated by the MTI system [25] which uses a combination of distributional and vocabulary-based approaches. Dimension reduction is accomplished as a part of the vector building process without involving computationally demanding techniques like SVD, which makes the method efficient and scalable. This approach has the additional advantage of being amenable to incremental updates. The reflective process used by the system is able to capture similarities in meaning between documents that do not share any common terms. This results in a better representation of the documents in the vector space as it addresses the problem of synonymy. Hence, this method may make a useful addition to techniques currently employed by the MTI.

Acknowledgments

We would like to thank Dominic Widdows for creating the Semantic Vectors Package and NLM for providing the MEDLINE database that facilitated our research.

References

[1] NLM – National Institutes of Health. 2009. Available from: http://www.nlm.nih.gov/bsd/licensee/2009_stats/2009_Totals.html.
[2] Schulman JL. NLM – National Institutes of Health. 2008. Available from: http://www.nlm.nih.gov/pubs/techbull/nd08/nd08_mesh.html.
[3] Kim W, Aronson AR, Wilbur WJ. Automatic MeSH term assignment and quality assessment. In: Proc AMIA Symp; 2001. p. 319–23.
[4] Névéol A, Mork JG, Aronson AR. Automatic indexing of specialized documents: using generic vs. domain-specific document representations. In: Biological, translational and clinical language processing. Prague; 2007. p. 183–90.
[5] Sohn S, Kim W, Comeau DC, Wilbur WJ. Optimal training sets for bayesian prediction of MeSH® assignment. J Am Med Inform Assoc 2008;15(4):546–53.
[6] Aronson AR, Mork JG, Lang FM, Rogers WJ, Neveol A. NLM medical text indexer: a tool for automatic and assisted indexing. Bethesda: U.S. National Library of Medicine; 2008.
[7] Cohen T, Schvaneveldt R, Widdows D. Reflective random indexing and indirect inference: a scalable method for discovery of implicit connections. J Biomed Inform 2009:14 [epub ahead of print].
[8] Kanerva P, Kristoferson J, Holst A. Random indexing of text samples for latent semantic analysis. In: Proceedings of the 22nd annual conference of the cognitive science society; 2000.
[9] Dasarathy BV. Nearest neighbor (NN) norms: NN pattern classification techniques. Los Alamitos: IEEE Computer Society Press; 1990.
[10] Yiming Y, Chute CG. An application of expert network to clinical classification and MEDLINE indexing. In: Proceedings of the eighteenth annual symposium on computer applications in medical care; 1994. p. 157–61.
[11] Yiming Y, Chute CG. A linear least square fit mapping method for information retrieval from natural language texts. In: 14th international conference on computational linguistics. Nantes; 1992. p. 446–53.
[12] Cooper GF, Miller RA. An experiment comparing lexical and statistical methods for extracting MeSH terms from clinical free text. J Am Med Inform Assoc 1998;5(1):62–75.
[13] Aronson AR. Effective mapping of biomedical text to the UMLS Metathesaurus: the metamap program. In: Proc AMIA Symp; 2001. p. 17–21.
[14] Aronson AR, Bodenreider O, Chang HF, Humphrey SM, Mork JG, Nelson SJ et al. The NLM indexing initiative. In: Proc AMIA Symp; 2000. p. 17–21.
[15] Humphreys BL, Donald LA, Schoolman HM, Barnett OG. The unified medical language system. Methods Inf Med 1993:281–91.
[16] Indexing Initiative. 2007. Available from: http://ii.nlm.nih.gov/Eval_Analysis/Eval_2007/summary.shtml.
[17] Hersh W. Information retrieval: a health and biomedical perspective. New York, NY: Springer; 2003.
[18] Salton G, McGill MJ. Introduction to modern information retrieval. McGraw-Hill, Inc.; 1886.
[19] Deerwester S, Dumais ST, Dumais DW, Landauer TK, Harshman R. Indexing by latent semantic analysis. J Am Soc Inf Sci 1990;41:391–407.
[20] Landauer TK, Laham D, Rehder B, Schreiner ME. How well can passage meaning be derived without using word order? A comparison of latent semantic analysis and humans. In: Shafto MG, Langley P, editors. Proceedings of the 19th annual meeting of the cognitive science society. Mawhwah, NJ: Erlbaum; 1997. p. 412–7.
[21] Kanerva P. Hyperdimensional computing: an introduction to computing in distributed representation with high-dimensional random vectors. Cogn Comput 2009;1(2):139–59.
[22] Sahlgren M. An introduction to random indexing. In: Proceedings of the methods and applications of semantic indexing workshop at the 7th international conference on terminology and knowledge engineering (TKE). Copenhagen, Denmark; 2005.
[23] Vempala S. The random projection method. American Mathematical Society; 2004.
[24] Johnson WB, Lindenstrauss J. Extensions of Lipschitz maps into a Hilbert space. Contemp Math 1984:189–206.
[25] Gallant SI. Context vectors: a step toward a "Grand Unified Representation". In: Hybrid neural systems, revised papers from a workshop. London: Springer-Verlag; 1998. p. 204–10.
[26] Schvaneveldt R, Cohen T. Abductive reasoning and similarity. In: Ifenthaler D, Seel N, editors. Computer based diagnostics and systematic analysis of knowledge. New York: Springer; 2010. p.189–211.
[27] Semantic Vectors hosted by Google Code. Available from: http://code.google.com/p/semanticvectors/.
[28] Lucene open source package. 2009. Available from: http://lucene.apache.org/.
[29] Martin DI, Berry MW. Mathematical foundations behind latent semantic analysis. In: McNamara LT, Sennis SW, editors. Handbook of latent semantic analysis. Lawrence Erlbaum Associates; 2007.
[30] Sitbon L, Bruza P. On the relevance of documents for semantic representation. In: Proceedings of the 13th Australasian document computing symposium. Hobart, Australia; 2008.
[31] Funk ME, Reid CA. Indexing consistency in MEDLINE. Bull Med Libr Assoc 1983;71(2).